ARI Pillar 1 — GEO Readiness & Governance

Full Technical Specification (v10.0)

This specification defines all machine-readiness, governance, identity, and trust requirements needed for AI agents to discover, access, interpret, trust, and legally interact with a website.

---

1. Purpose of Pillar 1

Pillar 1 ensures:

- Agents can **find** your site
- Agents can **crawl** it
- Agents can **understand** policies
- Agents can **verify** identity and trust
- Agents can **obey** licensing and legal terms
- Agents can **use** content safely and predictably

This pillar is the foundation of the entire ARI framework.

Major Sub-Components (10 Total)

Below is the complete specification for each.

1.1 — Sitemap Health & Freshness

Definition

The sitemap must accurately represent the structure of the site, provide valid URLs, and be machine-readable without errors.

Requirements

- A valid `sitemap.xml` must exist
- Must return **HTTP 200** with no parsing errors
- Must list only valid, reachable URLs
- Must be updated within last 30–90 days
- May reference multiple sitemaps

- GZIP support for `sitemap.xml.gz`
- No more than 5% broken URLs

Agent Checks

- Fetch sitemap index
- Validate XML
- Follow sitemap references
- Crawl and confirm each URL
- Compare against robots.txt directives

Failure Conditions

- Missing sitemap (critical fail)
- Sitemap returns 404/500
- Sitemap lists unreachable or blocked URLs
- Malformed XML

1.2 — Crawlability & Directive Integrity

Definition

Ensures robots directives do not block critical paths and are interpretable by modern agents.

Requirements

- Robots.txt must not block essential pages
- No contradictory rules for the same user-agent
- Consistent wildcard usage
- Crawl-delay must be reasonable (<5 seconds)
- No hidden sitemaps accidentally blocked

Agent Checks

- Parse robots.txt
- Confirm directive consistency
- Validate that sitemap URLs are not blocked
- Confirm `Allow` and `Disallow` logic

Failure Conditions

- Site fully blocked for all agents (critical)
- Contradictory rules
- Crawl delay > 10s

1.3 — AI Usage Policy (llm.txt)

Definition

A machine-readable file defining how AI agents may interact with, reuse, store, or reason over the site's content.

Requirements

- File located at `/.well-known/llm.txt` or `/llm.txt`
- Must include:
  - Allowed actions
  - Prohibited actions
  - Caching rules
  - Attribution requirements
  - API preference and rate limits
  - Commercial usage rules
  - Dataset extraction rules

Sample Structure

```
Version: 1.0
Allow: read, summarize
Disallow: training, embedding
Attribution: required
Cache-Window: 24h
Rate-Limit: 120/m
Commercial-Use: with-license
Preferred-API: https://example.com/api
```

Failure Conditions

- Missing llm.txt (major penalty)
- Unstructured text not following spec
- Contradictory rules

1.4 — Robots.txt Configuration

Definition

Traditional crawler governance file. Historically SEO; now critical for agent routing.

Requirements

- Must be accessible at `/robots.txt`
- Must include sitemap reference
- Should define per-agent logic if needed
- Must use valid syntax

Agent Checks

- Syntax validation
- Directive conflict detection
- Crawlability simulation

Failure Conditions

- Missing robots.txt (warn)
- Fully disallowed site (critical)
- Invalid syntax

1.5 — Canonicalization

Definition

Agents must know which URL is the "source of truth" when duplicates exist.

Requirements

- Every major page must include a `<link rel="canonical">`
- Canonical URL must be reachable
- Canonical chain must not loop
- Self-referencing canonical recommended

Agent Checks

- Validate canonical tag
- Fetch canonical target

- Compare content similarity

Failure Conditions

- Canonical to non-existent URL
- Circular canonicals
- Missing canonical tags on >40% pages

1.6 — Agents.json Implementation

Definition

A new standard defining the site’s complete machine governance model.

Requirements

- Located at `/.well-known/agents.json`
- Must be valid JSON
- Must include:
    - site metadata
    - owner identity
    - allowed agent actions
    - disallowed actions
    - API list
    - dataset availability
    - economic model
    - licensing
    - safety considerations

Sample Structure

```
{
  "version": "1.0",
  "owner": {
    "name": "Example Inc",
    "contact": "support@example.com"
  },
  "actions": {
    "allow": ["crawl", "read", "metadata"],
    "deny": ["batch-download", "training"]
  },
  "apis": ["https://example.com/api/v1"],
  "economic_model": "subscription",
  "license": "cc-by-4.0"
}
```

Failure Conditions

- Missing agents.json (major)
- JSON parsing errors
- Missing required fields

1.7 — Domain Trust Signals

Definition

Authenticates the site’s technical and governance identity.

Requirements

- Valid SSL certificate (≥30 days remaining)
- Correct SANs
- No mixed content
- DNSSEC recommended
- HSTS enabled
- Consistent www/non-www behavior

Agent Checks

- Parse certificate
- Validate expiry, SAN, and chain
- Identify redirect consistency

Failure Conditions

- Expired certificate (critical)
- Invalid domain mismatch

1.8 — Authorship Metadata

Definition

Ensures content has clear, verifiable authorship.

Requirements

- Article-Level author metadata
- `author`, `dateModified`, `publisher` in schema.org
- Organization Knowledge Graph linking
- Social verification optional

Agent Checks

- Extract author schema
- Validate timestamp freshness

Failure Conditions

- Missing author on majority of articles
- Invalid or contradictory metadata

1.9 — Economic Model Disclosure

Definition

Explicit declaration of how the site makes money, enabling agents to infer bias, trust vectors, and usage constraints.

Accepted Models

- Subscription
- Ads
- Affiliate
- SaaS/API billing
- Marketplace commission
- Donations
- Sponsorship
- Mixed

Requirements

- Must appear in `agents.json`
- Should appear in `llm.txt`
- Should appear in site footer metadata

Failure Conditions

- No economic model declared
- Contradictory monetization metadata

1.10 — Data Licensing

Definition

The legal usage terms for AI agents interacting with site content.

Allowed Models

- Proprietary / all rights reserved
- CC0
- CC-BY
- CC-BY-SA
- Custom license
- Commercial license required

Requirements

- Must be declared in `agents.json`
- Must not contradict llm.txt

Failure Conditions

- Missing license declaration
- Contradictory licensing

Scoring Model for Pillar 1

Each sub-component contributes:

- PASS: full points
- WARN: partial points
- FAIL: no points

Critical failures immediately drop the score to **≤10**.
